



TREBALL FINAL DE GRAU



ESCOLA
POLITÈCNICA SUPERIOR
UNIVERSITAT DE LLEIDA
INSPIRING THE FUTURE

Estudiant: FERRAN MARTÍNEZ PEDREÑO

Titulació: Grau en Enginyeria Informàtica

Títol de Treball Final de Grau: **Interfície Modular Muse:**
Disseny, construcció i testeig d'un sistema de captura i reinterpretació de senyals OSC

Director/a: **Jordi Virgili**

Presentació

Mes: Juny

Any: 2019

Index:

Agraiments.....	1
Bloc 1: Introducció.....	2
1.1 Muse, the brain sensing headband.....	2
1.2 Motivacions.....	4
1.3 Estructura del document.....	6
1.4 Objectius, riscos i costos.....	7
1.4.1 Objectius.....	8
1.4.2 Riscos.....	9
1.4.3 Costos.....	10
Bloc 2: Estat de l'Art.....	11
2.1 Marc contextual.....	11
2.1.1 Investigació.....	11
2.1.1.1 Muselab.....	11
2.1.1.2 Cartool.....	12
2.1.1.3 Bran Vision Recorder.....	13
2.1.2 Accessibilitat.....	14
2.1.2.1 Camera Mouse.....	14
2.1.2.2 HeadMouse.....	15
2.1.2.3 Brain2Speech.....	16
2.2 Disseny del Projecte.....	17
2.3 Particularitats del Projecte.....	19
2.4 Casos d'ús i públic objectiu.....	21
Bloc 3: Desenvolupament.....	22
3.1 Diagrama de Gantt.....	22
3.2 Problemes i solucions.....	22
3.2.1 Connexió Casc a PC.....	23
3.2.1.1 Enllaç a PC.....	23
3.2.1.2 Captura de senyals.....	24
3.2.2 Construcció Interfície Modular.....	25
3.2.2.1 OSCcontroler.....	26

3.2.2.2 ModularDispatcher.....	26
3.2.3 Construcció dels moduls.....	27
3.2.3.1 Experimental Module.....	27
3.2.3.2 Mouse Module.....	28
3.2.4 Construcció de l'Interfície Gràfica.....	30
3.3 Organització i Disseny.....	31
3.3.1 Sistema de fitxers.....	31
3.3.2 Diagrames de disseny.....	33
3.3.3 Versions i dependències.....	34

Bloc 4: Tests amb usuaris.....35

4.1 Tasques a realitzar.....	35
4.2 Metodologia.....	36
4.3 Resultats. Anàlisi Quantitatiu.....	38
4.4 Resultats. Anàlisi Qualitatiu.....	40
4.5 Observacions i dades demogràfiques.....	41
4.6 Conclusions.....	43

Bloc 5: Conclusions.....44

5.1 Expectatives vs Realitat.....	44
5.2 Punts forts i punts de millora.....	45
5.3 Treball futur.....	46

Bloc 6: Bibliografia.....47

Annex.....48

Index de Taules:

Taula 1. Temps d'execució de les tasques segons participant.	38
Taula 2. Mitjana d'execució de les tasques.	38
Taula 3. Puntuacions del qüestionari del producte.	40
Taula 4. Mitjanes del qüestionari del producte.	40

Index de Figures:

Figura 1. Image promocional del dispositiu Muse.....	3
Figura 2. Finestra de visualització bàsica de Cartool.....	13
Figura 3. Finestra de captura i usuari de BrainVision Recorder.....	14
Figura 4. Finestra de control i usuari de HeadMouse.....	15
Figura 5.Usuari interactuant amb Brain2Speech.....	16
Figura 6. Diagrama de Gantt de l'Interfície Modular.....	22
Figura 7. Captura de pantalla d'error.....	24
Figura 8. Organització del sistema de fitxers.....	31
Figura 9. Organització de la carpeta resources.....	32
Figura 10. Contingut de la carpeta scripts.....	32
Figura 11. Contingut de la carpeta src.....	32
Figura 12. Diagrama general del projecte.....	33
Figura 13. Diagrama de l'interacció entre classes GUI.....	34
Figura 14.Usuari utilitzant eViacam.....	35
Figura 15.Usuari utilitzant l'Interfície Modular.....	36
Figura 16. Explicació previa a un usuari abans de la sessió.....	37
Figura 17. Gossos confosos inclinant el cap.....	39

Agraïments

Si hagués de volcar tots els meus pensaments en un full en blanc crec que seria capaç d'omplir tres o quatre pàgines. Ho dic amb coneixement de causa, perquè anava per la segona quan he decidit tornar a començar. Així doncs, anem al grà!

Moltes gràcies a tots els professors que han contribuït a la meva formació. Gràcies per les tutories, per les sessions i per les pràctiques i els exàmens. Ara que ja estic acabant i entro en el món professional entenc millor la frase: *"Com més suïs en temps de pau, menys sangraràs en temps de guerra"*.

Moltes gràcies als meus tutors de TFG, a en Roberto García i a en Jordi Virgili. Gràcies per guiar-me i proporcionar-me solucions o directrius en els moments en que estava encallat.

Gràcies a tots aquells fets petits i inconexos que m'han portat a acabar aquesta carrera amb aquest treball. A vegades l'aleteig d'una papallona provoca tempestes.

I sobretot gràcies a la Mireia, el meu amor. Tú em vas impulsar a començar aquesta carrera i a seguir aquest camí. L'acabo per la meva mà però el vaig començar gràcies a tú. Només per tú.

A tots i a totes, moltes gràcies.

See you space cowboy...

Bloc 1: Introducció

Es pot descarregar el treball, el codi i tot el material multimedia generat en en el següent enllaç:
<https://drive.google.com/open?id=129ZuavZ97B0hLFhLLJXTojJ64leUVfy8>

Aquest treball neix amb l'objectiu de crear un sistema que actuï com a interfície per a les diferents senyals tant electroencefalogràfiques (EEG) com d'altres tipus, que emet un dispositiu de tipus Muse, capturant-les i permeten alhora una interpretació oberta d'aquestes, mitjançant agregacions que no modifiquen l'estructura del sistema base. Al sistema se l'ha anomenat Interfície Modular i a les agregacions Mòduls.

Per tal de demostrar l'eficàcia del sistema, s'han creat dos mòduls amb objectius ben diferenciats: ExperimentalModule i MouseModule.

El primer d'ells està orientat a l'ús en sessions d'experimentació amb usuaris on interressi guardar feedback dels seus senyals cerebrals. El segon entra dins de l'àmbit de l'accessibilitat i permet controlar totes els senyals d'input a l'ordinador provinents d'un ratolí (clics i moviment) al portador d'un casc Muse connectat a l'Interfície Modular.

A continuació es donaran detalls sobre el dispositiu emissor dels senyals: Muse headband, utilitzat en el projecte, s'explicaran els diversos motius que han portat a l'elecció d'aquesta temàtica de projecte i s'especificarà l'estructura que conforma el present document. A més a més, es ficaran en perspectiva els objectius, riscos i costos que ha comportat el projecte.

Però, primer de tot: què és Muse?

1.1 Muse, the brain sensing headband

Muse, és un dispositiu que brinda informació en temps real de l'activitat cerebral del seu portador. L'aparell té la forma d'una diadema acabada en forma de ganxo en els seus dos extrems, pensada per reposar sobre les orelles.

Al llarg de la seva estructura hi han incorporats múltiples sensors, que recullen tant informació magnética (per fer un electroencefalograma) del seu portador, com d'alguns gestos facials. A més a més, disposa d'un acceleròmetre, un aparell que proporciona dades de l'atracció del dispositiu en relació a la força gravitatòria terrestre.

L'objectiu amb el qual va ser creat Muse és el de permetre obtenir feedback de sessions de meditació per tal de saber-ne el rendiment. El feedback sol ser de tipus auditiu o visual i es presentat a l'usuari a través d'una aplicació mòbil desenvolupada pels mateixos creadors, que es connecta al casc via Bluetooth.

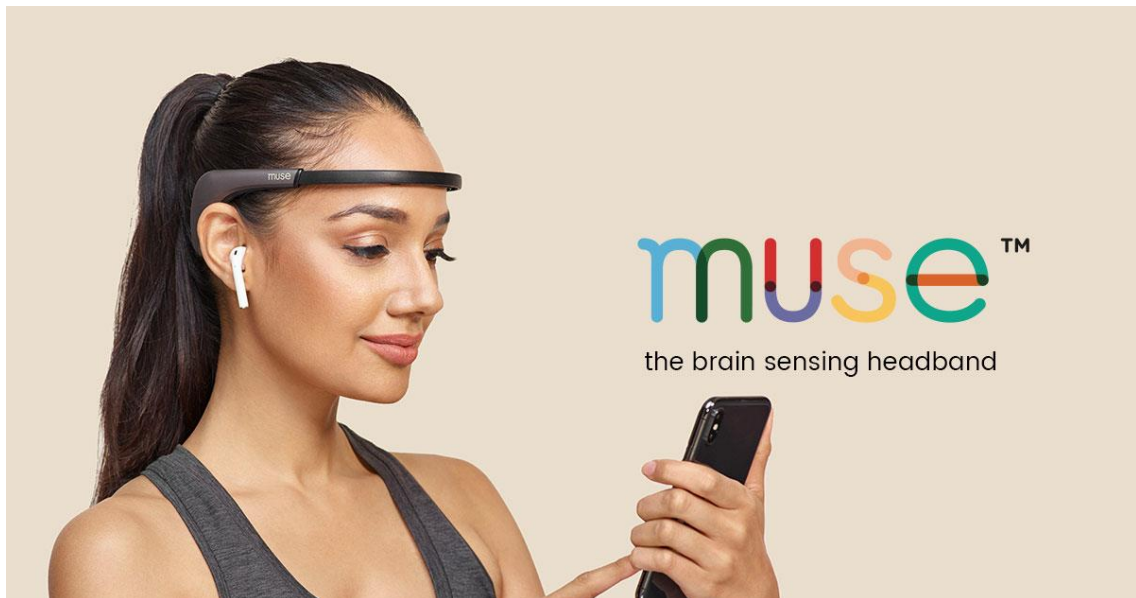


Figura 1. Image promocional del dispositiu Muse. Font:
<https://choosemuse.com/app/uploads/2018/10/muse-2-facebook-3.jpg>

La forma de feedback més popular és com a sons climàtics. Si el portador està relaxat, el clima serà calmat i plàcid i s'emetraran sons suaus; per contra, si el portador està nerviós o inquiet els sons canviaran als d'un clima més remogut i tempestuós. Adoptant aquesta forma, s'espera proporcionar un feedback compatible i útil durant una sessió de meditació.

El dispositiu va sortir a la venda al públic el Maig del 2014, però ja estava en desenvolupament des de l'any 2003. Els seus creadors són la companyia canadenca InteraXon Inc, empresa fundada l'any 2007 a Toronto per quatre socis: Ariel Garten, Trevor Coleman, Chris Aimone i Steve Mann.

Muse, va poder esdevenir una realitat gràcies a múltiples campanyes de crowdfunding. Actualment és un dispositiu força popular dins de múltiples àmbits, principalment el domèstic, però també dins l'àmbit científic i d'investigació.

Diverses universitats utilitzen Muse en projectes de recerca, entre elles la Universitat de Lleida (UDL), indret que albergava un d'aquests dispositius a l'inici d'aquest projecte.

1.2 Motivacions

Les corrents de la realitat ens tenen atrapats dins les seves urpes i ens bressolen implacables, ignorants als nostres desitjos i anhels. Davant les limitacions imposades pel nostre entorn i l'instant temporal que ens ha tocat viure, tenim dues opcions: acceptar-les o no.

La majoria accepta l'entorn i les eines existents que se li hi han atorgat sense ni tan sols plantejar-s'ho. És molt fàcil de fer, només cal viure el dia a dia com s'ha fet sempre. Altres atresoren el desig d'un canvi, però no tenen les eines, el coneixement o la voluntat per fer-lo efectiu. La resignació i l'esperança que algú altre els hi atorgui són de l'única via que disposen.

D'altres no es conformen. Estan inquiets, pensen, maquinen i busquen els recursos i el coneixement per fer realitat el seu pensament. Són aquells que modifiquen la realitat, que canvien el món. Són aquella gent que fa mil·lenis no van tenir por al foc, que van lligar pedres afilades a puntes de pals, que van considerar que carregar pes era millor sobre dues rodes i no sobre la seva pròpia esquena.

Pas a pas, espenta rere espenta, generació rere generació, la humanitat ha modificat la seva realitat i el seu destí. Hem passat de pintar els fons de les caveres amb els nostres dits a comunicar coneixement a través de satèl·lits orbitant la vasta fredor de l'espai.

Aquest treball pretén ser un minúscul gra de sorra dins de tot aquest procés d'expansió. Ha suposat fer realitat una idea, que ningú havia garantit que fos possible, en realitat.

A l'escollir una temàtica per al meu Treball de Final de grau (TFG), volia fer alguna cosa diferent, volia intentar tancar aquest cicle de quatre anys amb un treball ben fet, que em deixés bon sabor de boca.

Vaig saber de l'existència del dispositiu Muse dins l'UDL per part del professor Roberto Garcia, que ja m'havia comentat en alguna ocasió que existia un dispositiu semblant que controlava un Escalètric amb el pensament.

Rumiant-ho vaig considerar que un dispositiu així podia donar-se-li altres aplicacions si érem capaços de mapejar prou senyals. Així doncs vaig decidir basar el meu TFG en aquesta idea i veure on podia arribar.

Finalment he acabat amb un sistema que permet utilitzar-lo en investigació i fer més accessible la interacció amb l'ordinador. A més he creat una base per tal de que els senyals provinents del casc es transformin en tot allò que un desenvolupador sigui capaç d'imaginar.

La meva intenció ha estat crear quelcom nou del no-res. I que el resultat fos útil i la seva execució original.

En definitiva, la meva motivació principal ha estat canviar una mica el món. Empènyer una mica els límits del que és possible i el que no. I de pas empènyer una mica uns altres límits, no menys importants.

Els propis.

1.3 Estructura del document

- **Bloc 1:** Introducció

En aquest primer apartat es realitzarà una introducció general al projecte. Primerament es farà una breu descripció que resumirà l'abast del projecte i la seva temàtica.

A continuació s'entrarà en detall sobre el dispositiu utilitzat (Muse headband), es descriuran les seves característiques més importants, la seva funcionalitat principal i es donaran dades sobre els seus creadors i l'empresa que l'ha impulsat.

Després s'explicarà quines motivacions han portat a realitzar aquest treball i es farà una enumeració dels diferents blocs i una descripció del seu contingut. Finalment, es plantejaran els objectius i riscos que s'han hagut de marcar i tenir en compte en aquest projecte, així com d'una valoració econòmica sobre el cost total de desenvolupament que hagués suposat dins del mercat laboral.

- **Bloc 2:** Estat de l'art

En aquest segon bloc es tractarà el projecte des de diferents perspectives. Primerament es farà una anàlisi dels sistemes existents semblants dins del mateix àmbit. És a dir, una enumeració i descripció del software actual que interactua amb un dispositiu dins dels àmbits de la investigació i l'accessibilitat.

Un cop definit aquest marc contextual, es donarà una visió global del projecte pel que fa a disseny i s'establirà una relació amb l'entorn denotant que és el que aporta de nou i en què es diferencia.

Finalment es plantejarà quins casos d'ús pot tenir el treball i sobre quin públic objectiu pot estar dirigit.

- **Bloc 3:** Desenvolupament

Dins d'aquest bloc s'aprofundirà des d'una perspectiva de desenvolupador, en tots els detalls del procés de construcció del projecte.

Per una banda s'explicaran les diferents fases per les quals ha passat el treball. Es ficaran en perspectiva tots els problemes trobats en elles i com s'han resolt, s'aprofundirà en l'apartat tècnic de cada etapa i s'explicaran les principals decisions de disseny.

Per una altra, s'explicarà l'estructura del projecte des d'una perspectiva orientada a l'ús, les dependències de les llibreries utilitzades, com estan dissenyats els diferents components que formen el tot i l'organització dins del projecte i el sistema de fitxers.

- **Bloc 4:** Sessió d'usuaris

En aquest apartat es descriurà la sessió d'usuaris realitzada amb la Interfície Muse activada amb el mòdul d'accessibilitat. L'objectiu d'aquesta sessió va ser provar un dels mòduls amb usuaris reals, testejar-ne l'eficàcia i comparar l'efectivitat del producte amb d'altres de semblants. S'explicarà en què va consistir la sessió, les tasques a realitzar i els resultats, reaccions i comentaris dels participants.

- **Bloc 5:** Conclusions

En aquesta secció es farà una valoració en retrospectiva de tota allò que ha suposat el projecte i de l'estat en què s'ha deixat.

Es farà un seguit de valoracions sobre el que ha aportat el desenvolupament al seu creador i es farà un seguit de propostes sobre quines millores se li podrien fer al projecte, o cap a quina direcció es podria avançar.

- **Bloc 6:** Bibliografia

Finalment es farà una enumeració citant totes les fonts que han permès obtenir informació útil per a la realització del projecte.

1.4 Objectius, riscos i costos

Com en tot projecte o empresa que es planteja, per a la realització d'aquest treball s'ha hagut de marcar una sèrie de fites per triar una direcció sobre la qual centrar el treball. Alhora, el projecte ha suposat una sèrie de riscos que s'han hagut d'evitar i tenir en compte.

Finalment, tot treball té un cost, així que es fa necessari pensar en l'impacte econòmic que hagués suposat tot l'esforç de desenvolupament dins del mercat laboral actual.

A continuació s'explicarà un per un, els punts anteriors assenyalats.

1.4.1 Objectius

Per tal de definir correctament el sistema a crear, es va marcar tot un seguit d'objectius que aquest havia de complir.

Objectiu 1: El sistema ha de permetre capturar totes les senyals provinents del dispositiu Muse.

S'ha d'aconseguir mapejar correctament tota senyal provinent del dispositiu, així com deixar oberta la porta a la possibilitat de reconèixer-ne de noves.

Objectiu 2: El sistema ha de ser capaç de manipular els valors de les senyals capturades.

S'ha de poder operar de manera còmode i eficient amb els valors rebuts, per poder atribuir-los'hi una funcionalitat concreta.

Objectiu 3: El sistema ha de permetre diferents usos de les senyals sense malmetre o modificar l'estructura base de captura de senyals.

S'ha de poder reinterpretar les senyals de manera lliure i desacoplada dels components encarregats de capturar-les.

Objectiu 4: El projecte ha de contenir un component que permeti fer un ús de les senyals dins de l'àmbit de l'accessibilitat

S'ha de crear un mòdul que permeti fer un ús complet dels inputs d'usuari provinents del ratolí (clics i moviment)

Objectiu 5: El projecte ha de contenir un component que permeti fer un ús de les senyals dins de l'àmbit de l'investigació

S'ha de crear un mòdul que permeti capturar un subconjunt de les senyals segons pertoqui, guardar-les a un fitxer i alhora marcar diferents instants de temps desde una interfície d'usuari.

Objectiu 6: El sistema ha de permetre control desde l'interfície d'usuari sobre quin modul s'utilitza i sobre la configuració del mateix

S'ha de permetre elegir a l'usuari del sistema quin mòdul utilitza i ha de poder ajustar-lo dins la mesura del possible a les seves necessitats més concretes.

1.4.2. Riscos

De la mateixa manera, s'ha de tenir en compte quins poden ser els principals factors que poden ficar en perill el projecte.

El següent llistat de riscos és el que es va considerar més probable d'esdevenir.

Risc 1: Desconeixement parcial del llenguatge de programació

Al no dominar de manera global el llenguatge utilitzat (Python) pot ocorrer que certes idees de disseny, no es puguin dur a terme tal i com s'havien plantejat, fet que podria endarreir el procés de desenvolupament.

Risc 2: Desconeixença total de les característiques i detalls del dispositiu Muse

Desconeixent l'estructura i la funcionalitat de Muse, podia ser possible que el projecte no es pogués dur a terme si alguna característica l'hagués limitat. Per altra banda, podia ser possible que el dispositiu fos massa complex de tractar pel temps del qual es disposava per realitzar el projecte.

Risc 3: Carencia d'un sistema genèric per sincronitzar el dispositiu Muse amb un ordinador

Al no disposar inicialment d'un sistema per a integrar el casc amb un ordinador podia ser possible que aquest hagués de ser desenvolupat de zero, amb tot el que aixó comportaria.

Risc 4: Experiència nul·la treballant dins del desenvolupament de software amb dispositius físics

La possibilitat de trobar-se amb problemes específics del dispositiu físic era més que real. El risc requeria en que no es tenia experiència tractant incidències d'aquest tipus, fet que suposaria endarreriments i hores tractant problemàtiques possiblement trivials per algú amb més experiència.

1.4.3. Costos

Ara que ens trobem amb el projecte finalitzat, podem realitzar una valoració global del seu possible cost. Així doncs, calcularem el cost tenint en compte per una banda, la mà d'obra i per altra tots els costos dels diferents elements que han fet possible el projecte.

Començarem calculant la mà d'obra, fent una relació entre les hores invertides i el preu per hora d'aquestes.

La quantitat d'hores invertides en el projecte ha estat d'unes 190. Per altra banda considerarem que un programador junior cobra uns 18.000€ l'any. Això fa que aproximadament, el seu sou sigui de 1200 al mes. Si considerem que la jornada seria de 160 hores al mes, si dividim 1200 entre 160, obtenim que el preu per hora d'un programador junior és de: 7,5€ l'hora.

Així doncs tenint en compte les hores invertides i el preu per hora establert, el preu total de la mà d'obra és de: 1425€

Per altra banda, cal considerar el cost que ha suposat obtenir un dispositiu Muse. Actualment, el model que s'ha fet servir (Original Muse) costa uns 219€.

Finalment cal considerar la llicència d'ús de Pycharm, el entorn de desenvolupament triat per tirar endavant el projecte. El seu cost anual és de 199€. Com que el projecte no ha durat més d'un any aquest serà el cost que es considerarà.

Així doncs el total dels costos del projecte ascendeix a uns: 1843€

Bloc 2: Estat de l'Art

2.1 Marc Contextual

Una part ben important per al projecte és l'entorn que l'envolta. Per molt complex o enginyós que sigui, el seu valor real queda palès en aquell moment en què se'l col·loca en perspectiva amb altres productes o dissenys del mateix tipus.

Per tal de donar una visió amb perspectiva del projecte i el seu entorn, s'enumerarà i analitzarà diversos sistemes semblants que aportin funcionalitats del mateix tipus. Ja que per la seva naturalesa modular el projecte pot incloure's dins de diversos àmbits, se n'ha triat dues que actualment representen les àrees en què la Interfície Modular hi té major cabuda.

Aquestes àrees són: Investigació i Accessibilitat.

A continuació s'enumeraran i descriuran projectes de cada àmbit per servir de contrast amb l'explicació pel que fa a disseny del propi projecte.

2.1.1 Investigació

La captura i anàlisi de senyals dels camps elèctrics del cervell és una necessitat per a molts investigadors de diversitat de disciplines. A continuació es presenten tres exemples representatius dels tipus de sistemes existents actuals.

2.1.1.1 Muse-lab

Muse-lab és una aplicació Java creada pels propis desenvolupadors de Muse headband. L'aplicació en qüestió ha estat ideada amb la intenció de facilitar la interacció amb el dispositiu als programadors i/o investigadors que l'utilitzin.

.

L'aplicació és capaç de rebre tots els senyals provinents del dispositiu Muse a través de ports UDP i TCP, un cop es tingui el casc emparellat amb l'ordinador. Un cop en marxa i amb el casc sincronitzat, l'aplicació és capaç de mostrar la informació en forma d'una gràfica temporal on es mostren les oscil·lacions en els valors dels senyals en temps real.

Així mateix, l'usuari és capaç de triar i eliminar, mitjançant un sistema de checkboxes quins senyals es volen visualitzar. Alhora, permet emmagatzemar en un fitxer csv, tots els senyals marcats durant un període determinat.

Els valors principals d'aquesta aplicació són els de mostrar d'una manera gràfica i entenedora els valors dels diferents senyals i la possibilitat de guardar una sessió de captura per a un tractament posterior.

Muse-lab serveix doncs, com a porta d'entrada de dades, user-friendly i genèrica; essent possible utilitzar l'output generat pel programa per a tota mena d'objectius diferents.

2.1.1.2 Cartool

Cartool és una eina d'anàlisi de senyals EEG. És un sistema desenvolupat per la institució Functional Brain Mapping Lab (FBMLab), de Ginebra, Suïssa. El projecte ha estat construït en C++ i es va iniciar el 1996, actualment encara rep actualitzacions.

L'aplicació permet capturar, tractar i analitzar els senyals EEG de diferents dispositius en tota una varietat de formats de fitxer específics. També dona suport a l'anàlisi de senyals Event-related potentials (ERP), que són respostes cerebrals a esdeveniments sensorials, motors o cognitius específics. La particularitat principal del sistema és que basa la seva anàlisi dels camps elèctrics del cervell, no només en l'oscil·lació dels diferents senyals en brut sinó també de la seva posició relativa al voltant del cervell.

A més permet realitzar un pre-tractament del model de dades obtingut per tal de millorar la qualitat de les dades. La visualització d'aquestes pot prendre moltes formes, tan topogràfiques o en forma de mapa de calor.

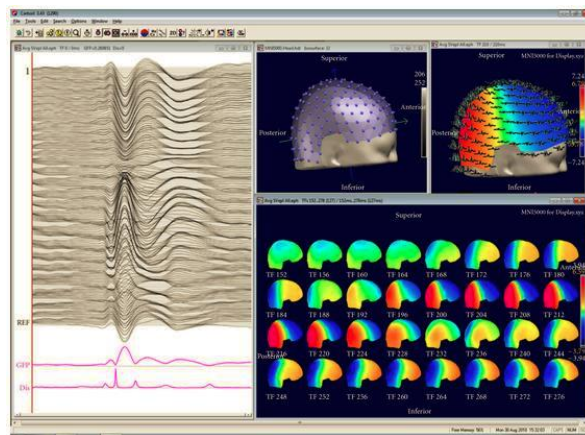


Figura 2. Finestra de visualització bàsica de Cartool. Font: <https://www.hindawi.com/journals/cin/2011/813870/fig3/>

L'entrada de dades es fa a través de fitxers d'informació raw de tipus EEG. Un cop dins el programa, aquest s'encarrega de fer tot l'anàlisi que haguem sol·licitat segons la configuració establerta.

Cartool permet doncs una gran versatilitat en el seus formats de representació gràfica i aporta a més un component topogràfic en relació a la generació de les senyals elèctriques cerebrals capturades.

2.1.1.3 Brain Vision Recorder

Aquesta tercera eina és un visualitzador de senyals. Ha estat desenvolupat dins de la marca Brain Products per l'empresa Bionic Ibérica S.A. amb seu a Barcelona.

Brain Vision Recorder permet visualitzar les dades independentment al seu procés de captura. Per aconseguir tal fita fa servir un sistema anomenat Remote Data Acces (RDA), alhora permet fer la captura i la visualització des de màquines diferents.

Permet a més fer lectures sobre els electrodes encarregats de fer la captura dels senyals EEG. Concretament és possible saber el grau d'impedància dels diferents canals mitjançant una presentació topogràfica dels elèctrodes juntament amb el seu elèctrode de referència assignat i la terra. Aquesta lectura es pot fer en qualsevol moment de la sessió de captura.



Figura 3. Finestra de captura i usuari de BrainVision Recorder. Font:

<https://www.hindawi.com/journals/cin/2011/813870/fig3/>

També permet seleccionar tots aquells canals i només aquells que interessin al seu usuari.

Així doncs Brain Vision Recorder deu la seva particularitat principal a ser un sistema de visualització dels senyals independent al seu procés de captura. Al ser independent permet a l'investigador executar-lo des d'una màquina diferent de la del procés de captura.

2.1.2 Accessibilitat

Existeixen diversos sistemes que supleixen la necessitat d'un control alternatiu del ratolí. L'esmentada necessitat prové principalment de persones amb mobilitat reduïda o amb necessitats especials.

2.1.2.1 Camera Mouse

Aquest sistema es basa en l'ús d'una webcam per al control del ratolí. La primera idea del sistema va sortir l'any 2000. El sistema original va ser creat pels professors: James Gips (Boston College) i Margrit Betke (Boston College).

Es va intentar comercialitzar el producte amb una start-up anomenada Camera Mouse Inc, per uns 395\$. Finalment la companyia va haver de tancar perquè no treia prou benefici.

La versió actual és una versió gratuïta ficada en desenvolupament per mà de Donald Green sota contracte de Boston College. La primera versió va sortir el 2007 i l'última el 2017.

El sistema permet fer un control del ratolí mitjançant el moviment del cap. El control d'aquest moviment es fa a través d'una webcam. Addicionalment el sistema permet fer clics esquerres movent el cursor a un punt determinat. Aquesta opció ha de ser activada en les configuracions.

CameraMouse és un sistema basat en el moviment del cap per a controlar el ratolí. Funciona bé, segons els seus propis desenvolupadors, en tasques que no requereixen una precisió molt elevada i que només requereixin clics esquerres.

2.1.2.2 HeadMouse

HeadMouse és un altre sistema de control del ratolí basat en l'ús d'una webcam. Ha estat desenvolupat per la Universitat de Lleida en col·laboració amb Indra, mitjançant la fundació Adecco.



Figura 4. Finestra de control i usuari de HeadMouse. Font: http://robotica.udl.cat/catedra/headmouse/manycam/hm_manycam.jpg

El sistema utilitza els moviments del cap per controlar el cursor i és capaç de permetre realitzar clics a l'usuari a través de dos mètodes. El primer es basa en l'ús d'un menú de clics que s'obra en el punt en què interessa interactuar-hi i deixa triar quin clic és l'intencionat.

El segon dels mètodes fa ús de macros per associar moviments facials a clics concrets.

Originalment el sistema va ser creat pensant a aprofitar al màxim les prestacions de Windows XP, sent la seva data d'inici al voltant del 2008. HeadMouse actualment es troba en la versió 4.4, versió que va sortir el 2016.

Durant les darreres s'ha fet èmfasi en el desenvolupament de millores en la compatibilitat amb els nous sistemes operatius de Windows i amb les webcams dels portàtils. Actualment, el seu manteniment està aturat i no es planeja més actualitzacions.

2.1.2.3 Brain2Speech

És un sistema basat en l'ús del casc Muse, que permet parlar a través dels senyals generats pel dispositiu. El projecte intenta proporcionar a les persones paralizades i incapaces de parlar, un mitjà per comunicar-se.

Per tal d'emetre sons, l'usuari es val dels seus propis moviments facials (el parpelleig i la contracció muscular de la cara). Un cop es reben aquests senyals, el sistema es val d'una graella on hi ha encabit tot l'alfabet i on l'usuari selecciona lletra per lletra la paraula que vol.



Figura 5. Usuari interactuant amb Brain2Speech. Font:

<https://github.com/Samin100/Brain2Speech/raw/master/brain2speech.gif>

La metodologia es basa en la navegació per la graella, primer per columnes i després per files. Un cop s'ha construït la paraula desitjada, l'usuari navega fins a la cel·la on fica END per indicar-ho. Després el sistema reproduïx la paraula mitjançant un sintetitzador de veu.

El projecte està construït en python i pensat per a navegadors. Utilitza també un servidor OSC i un dispatcher per a donar sentit als valors rebuts del casc.

2.2 Disseny del Projecte

L'Interfície Modular és un software capaç de capturar i reinterpretar els senyals generats pel seu portador amb la intenció de permetre utilitzar el casc per a usos diversos. A causa de la naturalesa de la seva estructura, la Interfície permet escalar els usos actuals implementats sense modificar el seu nucli.

En el procés de desenvolupament hem estat capaços de crear usos significatius per aquest dispositiu en dues àrees ben diferenciades: la investigació i l'accessibilitat.

L'ús implementat dins de l'àrea de la investigació permet recollir dades a temps real del portador del casc durant una sessió d'experimentació. El software permet a l'investigador triar els senyals que li interessa del seu portador, des de la interfície gràfica, i bolcar-les en un arxiu csv.

A més, permet marcar punts d'interès durant el recull de dades des de la interfície gràfica, que indicarien esdeveniments d'importància produïts durant la sessió. Aquests punts d'interès es poden canviar de nom durant la sessió i també es veuran reflectits dins l'arxiu de captura.

Alhora cada senyal estaria acompanyada d'una marca de temps, per a tenir un control exhaustiu sobre el moment en què s'ha recollit.

L'utilitat creada dins de l'àmbit de l'accessibilitat consisteix en un sistema d'Interacció-Persona-Ordinador que permet al seu usuari controlar el ratolí, aprofitant sensors de posició i electromiogrames (per detectar la musculatura facial) integrats en el casc. L'usuari és capaç de desplaçar el ratolí per la pantalla a través dels moviments del cap. A més pot fer clic, doble clic i clic dret mitjançant moviments musculars facials i parpelleig.

El mateix usuari a més pot modificar des de la GUI diversos paràmetres del control del ratolí, com la sensibilitat, l'interval de captura de senyals o la quantitat d'accions musculars necessàries per a realitzar una acció.

Per tal de permetre utilitzar el casc per a usos tan diversos, l'arquitectura del software s'ha implementat des de bon començament, amb aquesta idea. L'esmentada arquitectura, està formada per dos tipus d'elements, ben diferenciats.

Per una banda, s'ha desenvolupat una estructura que serveix de base per capturar els senyals del casc i preparar-les per a ser fàcilment accessibles, tractables i interpretables per agregacions que

compleixin uns criteris determinats definits. A tota aquesta estructura l'hem anomenat Interfície Modular.

D'altra banda s'han creat diverses agregacions que actuen sobre els valors dels senyals capturats i els reinterpreten per donar-los-hi diferents utilitats. A aquestes agregacions els hem anomenat Mòduls.

Tots aquests components de software han estat desenvolupats en Python per tal de ser independent del sistema operatiu que utilitzi l'usuari.

retar tots els senyals osc EEG i EMG, provinents del casc Muse a través d'una Interfície Modular.

Tot I que actualment nomès consta de dos moduls el sistema no tanca la porta a diferents usos dels ja construïts gràcies a la seva estructura, fet que permet escalabilitat i tota mena de possibilitats.

2.3 Particularitats del Projecte

Analitzades les eines dels sectors on incideix aquest projecte, seguidament analitzem el que aporta i quines particularitats són les més representatives dins del marc que l'envolta.

Primerament, centrant-nos en l'àmbit de la investigació, el mòdul Experimental veiem que és bastant semblant als sistemes que hem exposat en l'apartat anterior.

Aquest mòdul és capaç de capturar i visualitzar per terminal tots els valors obtinguts a través del casc. No presenta els valors d'una manera gràfica però permet guardar-los en un output de format estàndard, tal com fan la majoria d'aquestes aplicacions.

Una avantatja respecte a algun dels sistemes és que permet fer un ús i manipulació de les dades a temps real. Per contra, en altres sistemes es fa necessari realitzar primerament la sessió per després tenir impacte sobre les dades.

El mòdul Experimental permet incidir sobre les dades a través d'esdeveniments controlables per l'investigador. Això permet fer un pretractament de les dades previ a la seva anàlisi, fet que assegura un estalvi de temps i feina dins d'un procés d'investigació.

A més, també assegura un estalvi econòmic pel dispositiu de captura de dades triat. El dispositiu Muse és sensiblement més barat que la majoria d'aparells de captura electroencefalogràfica disponibles actualment al mercat.

Per altra banda, dins del món de l'accessibilitat, veiem que el projecte s'assembla en funcionalitat al seu entorn, però poc en execució:

La majoria de sistemes de control alternatiu del ratolí es basen en l'ús de la webcam. Això proporciona certes avantatges: no cal comprar cap dispositiu extra (almenys en aquells ordinadors que tinguin webcam integrada), no cal cap ajustament de sensors i el mètode d'interacció no és invasiu per a l'usuari. Però aquests sistemes no estan exempts d'inconvenients: la precisió en algun d'ells no és massa bona, depenen molt de la imatge (per tant, la il·luminació i l'entorn poden provocar interferències en el control) i en ser genèrics per a tota mena de webcams provoquen que siguin fàcilment vulnerables a incidències de compatibilitat segons el dispositiu o el sistema operatiu que les gestioni.

En canvi, utilitzant l'Interfície Modular, no depenem d'una bona il·luminació o l'entorn. Els valors dels senyals sempre són els mateixos independentment d'aquests factors. Per altra banda, el mòdul creat és bastant precís; permet una navegació concreta i una incidència en elements molt petits.

A més la gestió dels clics és customitzable i és completa. Aquesta gestió a més, mapeja aquestes senyals amb moviments naturals de la cara de l'usuari.

Finalment el punt més representatiu del sistema és la seva versatilitat. L'arquitectura modular permet fer incidència en moltes àrees alhora, deixant que una mateixa senyal actuï de forma particular en dos mòduls diferents. Aquesta característica diferencia el projecte de tots els altres sistemes, que només serveixen a un propòsit determinat.

A més, aquesta fet cobra especialment rellevància actualment, ja que InteraXon INC està deixant de banda el suport al desenvolupament per al dispositiu Original Muse. L'empresa ha decidit apostar per Muse 2, el seu nou dispositiu i està promovent que els developers facin servir Muse Direct, una aplicació de pagament que actua igual que el driver utilitzat per sincronitzar el casc amb el PC. També està ficant traves al lliure accés a l'esmentat driver.

Així doncs, el dispositiu Original Muse s'està deixant de banda per part dels seus desenvolupadors i un projecte com aquest pot servir de base perquè el desenvolupament amb aquest dispositiu continuï essent accessible.

2.4 Casos d'ús i públic objectiu

Com a secció final d'aquest segon bloc ens centrarem en analitzar els principals casos d'ús dels que pot ser protagonista el sistema i del públic que hi participaria.

Donades les seves principals característiques i funcionalitats, s'ha establert que la Interfície Modular podria generar els següents tres casos d'ús:

Cas d'ús 1: Recull de dades a nivell d'investigació.

Mitjançant el mòdul experimental, el sistema podria ajudar al recull de dades dels camps elèctrics del cervell dins de sessions d'investigació amb usuaris reals. El format de sortida del sistema permetria un tractament fàcil de les dades i l'ús de Muse com a dispositiu de captura permetria abaratir els costos.

Públic objectiu: *Investigadors, doctors.*

Cas d'ús 2: Dispositiu de control alternatiu del ratolí.

A través del mòdul de control del ratolí, es podria permetre un control mitjançant el dispositiu Muse a tot aquella persona que presenti necessitats específiques que causin que el disseny físic de control del cursor no sigui el més adequat per a elles o que danys diversos impedeixin l'ús de la webcam.

Públic objectiu: *Persones amb necessitats especials, persones amb paràlisi total o parcial.*

Cas d'ús 3: Sistema base per a aplicacions basades en l'ús de Muse

Utilitzant els estàndards de l'Interfície Modular, un desenvolupador seria capaç de crear un nou mòdul compatible amb aquest component que dotés d'una nova interpretació als senyals del casc.

Públic Objectiu: *Desenvolupadors de software.*

Bloc 3: Desenvolupament

El valor d'un producte no és només el producte en si, tot l'esforç necessari per construir-lo n'és una bona part. Per tal que tot aquest esforç no sigui en va, cal una planificació acurada i diverses fases de desenvolupament.

Un producte complex, genera problemes complexos. Però per tal de solucionar-los no hem d'aspirar a resoldre'ls directament. Si es vol tractar un problema complex s'ha de dividir en problemes simples, atacar-los individualment i sumar paulatinament el seu valor al producte final.

3.1 Diagrama de Gantt

Per tal de planificar el projecte i identificar les diferents parts i seccions a treballar individualment, s'ha creat el diagrama de la figura 6.

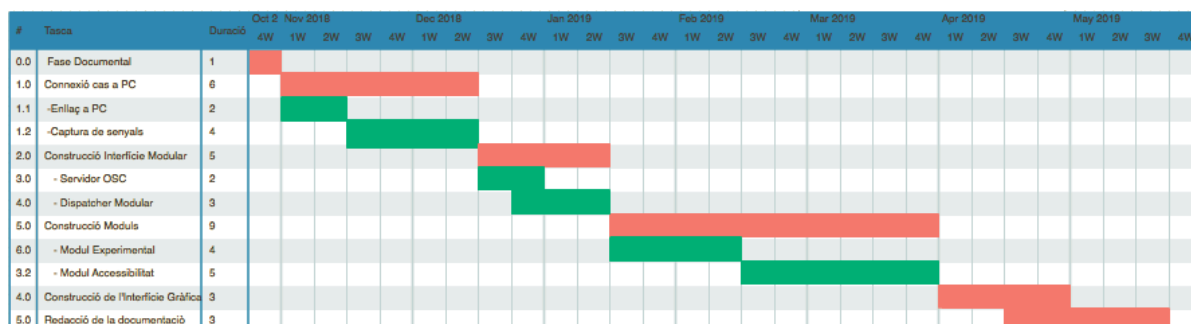


Figura 6. Diagrama de Gantt de l'Interfície Modular. Font: Elaboració propia

Així doncs veiem que el procés de desenvolupament ha constat de 5 grans fases, aquests són: la connexió cas a PC, la construcció de la Interfície Modular, la construcció dels mòduls, la construcció de la Interfície Gràfica i per últim i no menys important, la redacció de la documentació.

3.2 Problemes i solucions

En els següents apartats, tal com s'ha fet durant el procés de desenvolupament s'atacarà cada fase descrita en la figura 5 per separat, en l'ordre en què s'ha dut a terme i s'exposaran els principals problemes que s'han esdevingut així com les solucions escollides per resoldre'ls.

A més a més, es donarà detalls sobre les principals decisions de disseny i sobre els seus motius. No s'explicarà el punt 5, redacció de la documentació, ja que es tractaria d'explicar el treball redactant aquest document, el que seria un exercici recursivament redundant.

Finalment, en l'última secció d'aquest bloc, es donarà una visió global del projecte a escala tècnica, fent èmfasi en l'estructura de classes, per tal de donar una imatge clara de les diferents parts a un possible desenvolupador que prengué el producte i volgués entendre'l en la seva totalitat.

3.2.1 Connexió Casc a PC

Un dels primers passos i un dels més fonamentals va ser aconseguir sincronitzar el dispositiu Muse amb un ordinador i capturar-ne els senyals. La idea del projecte no es podria dur a terme si no s'aconseguia aquest pas. Per tal d'aconseguir l'objectiu es va haver d'assolir cada pas per separat, en format de seqüència, començant per enllaçar el casc a l'ordinador.

3.2.1.1 Enllaç a PC

Afortunadament, no va ser especialment costos trobar una eina per fer la connexió. Realitzant una mica de cerca per la web oficial de Muse, es va trobar un toolbox amb diversos programes per facilitar el procés. Aquest paquet s'anomena: *muse-research-tools* i inclou dos executables i un aplicatiu.

Concretament, el primer dels executables (*muse-io*) és una eina per actuar com a *driver* i enllaçar el casc a un port UDP o TCP; el segon (*muse-player*) permet escoltar el port UDP o TCP desitjat i visualitzar les dades per consola.

L'aplicatiu (*muse-lab*) també permet escoltar les dades provinents d'un port i a més a més exposar-les en una funció de temps segons el senyal o conjunt de senyals que escollim mitjançant una interfície gràfica.

Tot aquest paquet d'eines és compatible amb Mac i Linux, per als usuaris de Windows és necessari un altre paquet o utilitzar Muse Direct, una eina de pagament.


Resumint, el paquet *muse-research-tools* contenia:

- *muse-io*: Executable amb funció de driver que permet enviar totes les senyals de Muse per un port UDP o TCP
- *muse-player*: Executable que actua com a listener d'un port i mostrar les dades rebudes per consola.

- muse-lab: Aplicatiu que permet escoltar les dades d'un port concret i mostrar gràficament aquelles escollides mitjançant una sistema proporcionat per una interfície gràfica.

Així doncs, es va decidir aprofitar muse-io per a realitzar la connexió a pc. Gràcies a aquest driver es podia reencaminar totes els senyals per un port d'un protocol bàsic de comunicació. La part més desconeguda quedava solucionada i ja es començava a entrar en terrenys més familiars.

Malgrat això, la principal dificultat dins d'aquesta fase es va esdevenir a l'hora de crear el projecte i moure l'executable dins d'un subdirectori. En provar d'executar llavors l'script, la consola mostrava el següent error:



```
dyld: Library not loaded: /usr/local/lib/liblo.7.dylib
Referenced from: /Applications/Muse/.muse-io
Reason: Image not found
```

Figura 7. Captura de pantalla d'error. Font: Elaboració propia

El problema es generava quan l'script provava de trobar un parell de llibreries en la ruta mostrada en la imatge i no trobava res. Per solucionar-ho simplement es van copiar les llibreries (incloses dins del toolbox) al directori indicat per l'error i el problema es va solucionar.

3.2.1.2 Captura de senyals

El següent pas, un cop executada la fase anterior, era aconseguir capturar els senyals de manera que fossin fàcilment tractables. L'objectiu doncs, era incloure-les dins d'un script de Python per tal de començar a manipular-les segons s'esdevingués.

Aquesta sub-fase va ser de les més difícils, a causa de la desconexió dels mecanismes existents de captura i del format dels senyals emeses pel dispositiu Muse.

El primer enfocament que es va triar va ser intentar tractar els senyals a escala de byte. Aquest enfocament va ser recomanat per la mateixa pàgina de Muse orientada als desenvolupadors. Concretament, es recomanava l'ús de Protocol Buffers, un projecte Open Source de Microsoft. A més, es proveïa de diverses eines per fer més fàcil la integració al programa del desenvolupador, tot orientat a diferents llenguatges (entre ells python).

Per tal de fer servir Protocol Buffers calia instal·lar un compilador especial que reconegués el format descriptiu utilitzat per aquest projecte, per indicar l'estructura que havien de prendre les dades binàries un cop traduïdes a informació natural.

Un cop instal·lat, a més a més, calia escriure un fitxer on s'indiqués el format concret que s'atribuïa als nostres senyals; per sort, aquest fitxer ja estava a disposició del públic des de la mateixa pàgina de Muse.

Així doncs, es va instal·lar el compilador i es va incloure el fitxer dins del projecte; però a l'hora d'integrar-ho amb el codi es va trobar que les llibreries Python provocaven conflictes a l'hora d'importar-les. Més tard, es va descobrir que aquest projecte estava pensat per a Python 2, no per a Python 3 (la versió que s'ha escollit per al treball); aquest era l'origen del conflicte.

Com que es va optar per no fer el downgrade a Python 2, es va decidir destriar Protocol Buffers com a solució.

Finalment, la solució va sorgir per una altra via. Seguint el consell d'un dels cotutors, es va optar per provar d'utilitzar un servidor osc per tal d'intentar capturar els senyals utilitzant aquest protocol. Analitzant millor la documentació de Muse, es va veure que els senyals eren emeses d'aquesta forma.

Així doncs, un cop es va veure que aquesta via era possible, es va utilitzar una llibreria de Python anomenada *pythonosc* per tal de fer la connexió. Es va triar aquesta llibreria en concret, per la claredat del seu ús i documentació i perquè utilitzava UDP, un dels dos protocols amb els quals es podia enllaçar mitjançant el driver muse-io.

3.2.2 Construcció Interfície Modular

En aquesta segona fase, va tractar de crear la base de l'aplicació. Es van erigir els pilars fonamentals del concepte que acabaria esdevenint l'Interfície Modular Muse. Va ser una fase delicada; un error en el disseny dins d'aquesta fase, podria provocar greus dificultats en fases posteriors.

Inicialment, el primer enfocament considerava l'opció d'integrar la responsabilitat d'iniciar un servidor OSC i d'implementar mecanismes per a la manipulació de les senyals, dins d'una mateixa classe.

Després però, es va veure que aquest enfocament complicava molt el codi i el funcionament d'una instància d'aquesta hipotètica classe, anomenada llavors EEGcontroler. A més a més, aquest disseny violava un dels principis S.O.L.I.D; el SRP (*Single Responsibility Principle*), motiu més que suficient per reconsiderar el disseny.

Així doncs es va optar per un patró de disseny que separés responsabilitats i fes més senzill el codi i més clares les capacitats i funcionalitats de les classes que havien de gestionar aquesta part.

El resultat van ser dues classes: OSCcontroler i ModularDispatcher.

3.2.2.1 OSCcontroler

La responsabilitat assignada a aquesta classe és la d'iniciar un servidor UDP que accepti el format de senyal OSC. Per tal d'instanciar-la cal passar com a paràmetre una instància de ModularDispatcher, juntament amb un host i un port. Aquest dos últims paràmetres són opcionals, ja que existeixen dos valors per defecte: '127.0.0.1' i '5000'.

S'ha optat per obligar a passar una instància de ModularDispatcher pel constructor per tal que no es pugui iniciar un servidor sense indicar una interpretació dels senyals rebudes. Aquesta interpretació és tractada i assignada a través de ModularDispatcher, la classe que actua verdaderament d'interfície dins del projecte.

3.2.2.2 ModularDispatcher

És probablement la classe més important i la que otorga la principal de les seves qualitats al projecte. La modularitat.

Aquesta classe va ser construïda pensant en el cinquè dels principis S.O.L.I.D, el DIP (*Dependency Inversion Principle*). La intenció va ser crear una estructura que permetés aprofitar al màxim els senyals emeses pel dispositiu. Es podria haver optat per directament crear un programa 'a mida' i crear un aplicatiu que fes la funcionalitat d'un dels mòduls; però gràcies a aquesta classe podem donar el sentit que vulguem als senyals gràcies als seus mètodes d'interfície i als mòduls.

Concretament, la 'màgia' s'esdevé gràcies al constructor i a tres mètodes: configure, mapPath i mapSameFunctiontoPaths.

Per tal d'instanciar la classe, cal passar per constructor una instància modular. Perquè una classe sigui considerada modular ha d'implementar un mètode anomenat configure i que accepti un Modular Dispatcher com a paràmetre. Dins d'aquest mètode el mòdul podrà configurar el dispatcher, utilitzant les funcions de la classe ModularDispatcher: mapPath i mapSameFunctiontoPaths, per donar un sentit als senyals segons correspongui al mòdul.

Així doncs, per aquest motiu, ModularDispatcher és el component més important de tota l'estructura de classes que formen el projecte. Gràcies a l'anterior estructura, el programa és escalable en funcionalitats fins allà on es vulgui arribar construint mòduls.

3.2.3 Construcció dels Moduls

Un cop fonamentada la base, ja era possible començar a construir diferents interpretacions dels senyals. Per tal de demostrar que l'estructura del projecte funciona, es va decidir construir dos mòduls de funcionament ben diferenciat: `ExperimentalModule` i `MouseModule`.

Així mateix, com a prova inicial, també es va crear un mòdul anomenat `printModule` que es limita a mostrar per consola els senyals. Aquest mòdul està inclòs dins del projecte, però no és accessible dins de la GUI; no s'explicarà en detall dins d'aquesta secció a causa de la seva trivialitat (encara que per al desenvolupador pot esdevenir un bon *minimum working example*).

3.2.3.1 Experimental Module

Aquesta classe aprofita les bondats de l'estructura creada per `ModularDispatcher` i crea un sistema per a capturar els senyals i bolcar-los a un fitxer csv. El mòdul guarda cada senyal rebut, juntament amb una marca de temps i es capaç de senyalar punts d'interès dins del fitxer mitjançant un sistema d'esdeveniments personalitzable, controlable des de la GUI pel mateix usuari.

Els usuaris als quals va dirigit aquest mòdul serien investigadors, i podrien utilitzar el sistema per a capturar les fluctuacions en els senyals elèctriques cerebrals de subjectes en sessions d'usuari. Podrien, des de la pròpia GUI, triar només aquells senyals que interessa capturar i marcar punts d'interès (que després ajudarien a l'hora d'identificar moments concrets dins de l'experiment).

La seva estructura de mètodes de classe, segueix la marcada per les pautes necessàries per interactuar amb `ModularDispatcher`. Implementa el mètode `configure`, que assigna un handler a cada senyal mitjançant el mètode `mapSameFunctionToPaths` a les rutes osc indicades pel constructor d'instància.

El handler és `saveResults` i volca el senyal dins del fitxer csv esmentat. Aquests fitxers finalment es creen, un cop acabada la sessió, dins d'un directori dins del projecte anomenat `signals`. A més a més, inclou un mètode per canviar l'estat de l'esdeveniment, per a poder marcar punts dins de la sessió.

A nivell de desenvolupament, va ser el mòdul que va donar menys feina. Principalment, les dificultats van sorgir a l'hora de crear la seva GUI.

Un altre assumpte que es va haver de tractar, va ser la mida dels fitxers csv generats per les sessions de captura; hi havia vegades que en pocs minuts es generaven fitxers de diverses Mbytes.

Per pal·liar aquest problema es va introduir el mínim de dades al fitxer per senyal per tal de reduir el seu pes.

3.2.3.2 Mouse Module

Seguint també l'estructura d'interfície de ModularDispatcher, aquesta classe reinterpreta els senyals per convertir-los en un sistema d'accessibilitat, que proporciona un control alternatiu del ratolí.

Concretament aprofita tres senyals del dispositiu: muse/acc , muse/elements/jaw_clench , muse/elements/blink.

Per una banda, el senyal muse/acc prové d'un acceleròmetre intern, un aparell que mesura canvis en l'atracció del cos a la terra, per cada eix. Un acceleròmetre en caiguda gravitacional lliure cap al centre de la Terra mesuraria un valor de zero. Aquest senyal s'utilitza dins del mòdul per a permetre el moviment del ratolí mitjançant el moviment del cap.

Per altra banda, el senyal muse/elements/jaw_clench prové de les pertorbacions en els senyals elèctrics que detecten els EEG, electromiogrames (EMG), i permet reconèixer moviments facials, com arrugar el front o tensar la galta. Dins del mòdul es fa servir per a realitzar els clics.

Finalment, la senyal muse/elements/blink prové també dels sensors del dispositiu i reconeix el parpadeig. Es fa servir dins del mòdul per a realitzar el clic dret.

Respectivament, la interpretació de les senyals es fa a través de tres handlers: move, click i blink. Dins de cadascun hi ha la rutina necessària per a fer cadascuna de les tres accions abans descrites.

Per aconseguir-ho s'ha hagut de fer un treball exhaustiu, analitzant els senyals disponibles i els rangs generats i després traduint els valors a accions d'input d'usuari. També s'ha hagut d'utilitzar una llibreria anomenada pyautogui, per a controlar els inputs des de python. Ha estat la part més difícil i que més problemes ha generat.

Inicialment es va començar probant de moure el ratolí. Va ser relativament fàcil generar moviment, però no moviment acceptable i còmode per a un usuari.

En les primeres proves, el ratolí avançava a batzegades sense una animació clara de continuïtat. És a dir, si se li indicava al ratolí que es mogués 10px, el ratolí ho feia, però desapareixent i apareixent en la nova posició contínuament. Aquest fet generava molta incomoditat visual.

A més a més, en les primeres versions de moviment, aquest estava relacionat amb la mida de la pantalla. Verticalment, el moviment era bastant ràpid i mitjanament precís, però horitzontalment, no es generaven prou senyals per a mapejar correctament la pantalla. A més això podia generar problemes en pantalles de diferent mida.

També teníem el problema afegit de què el ratolí no s'aturava mai. En definir un handler de moviment pels senyals de l'acceleròmetre, el ratolí es movia, però no hi havia cap forma de fer-lo parar.

Tots aquests problemes van anar-se solucionant a poc a poc.

Per començar es va investigar en més profunditat la llibreria utilitzada pyautogui i es va esbrinar que existien flags d'animació assignables al mètode de moviment. És a dir, podíem indicar a la llibreria que simules una continuïtat del ratolí entre moviments de posició a posició.

Respecte al moviment general, es va destriar la idea de mapejar la pantalla i es va triar crear un sistema de moviment relatiu a la posició actual del ratolí. Segons el moviment del dispositiu, el ratolí avançaria cap a una direcció concreta respecte a la posició en què es trobés.

Es va decidir també disminuir la velocitat en què es movia el cursor, ja que impedia fer-ne un ús precís. Cal senyalar també que el cursor, un cop triada la direcció avançaria autònomament sense aturar-se, de tal manera que no caldria girar excessivament el cap per avançar, simplement triar la direcció fent un lleuger gest cap a aquella desitjada.

Finalment, per tal d'aturar el cursor de forma natural, es va decidir crear una zona d'aturada. Aquesta zona seria determinada per la posició inicial en què s'iniciés el mòdul. D'aquesta forma, l'usuari podria parar el cursor simplement adoptant aquesta posició. Aquesta característica, unida al sistema de moviment explicat en l'anterior paràgraf està pensat per a persones amb mobilitat reduïda, que no puguin moure excessivament el coll.

Per altra banda també van sorgir problemes en fer els clics. Principalment van sorgir en assignar als senyals el seu handler.

En el cas del botó esquerre, quan reconeixia el senyal, no era única; normalment en arrugar el front es generaven cinc o sis senyals positius, fet que provocava sis clics gairebé simultanis. A més, també calia trobar una manera de permetre realitzar el doble-clic dins d'aquesta situació.

El cas del botó dret, també provocava un problema semblant i era que s'activava el botó dret inesperadament pel parpelleig natural del portador del dispositiu. Per solucionar els problemes del

clic esquerre, es va optar per destriar els següents senyals rebuts per evitar clics innecessaris. A més es va crear un sistema d'interval, per aconseguir fer doble-clic si es tornava a rebre una altre senyal dins d'un temps determinat.

Finalment, es va aconseguir solucionar els clics drets innecessaris provocant que la finestra del botó dret no s'obris fins que rebés un cert nombre de senyals de parpelleig (per defecte 4) dins d'un interval determinat de temps (1 segon).

Per acabar, senyalar que també es van crear diversos getters i setters per a poder configurar certs paràmetres des de la GUI, pel mateix usuari. Entre ells: la sensibilitat de la zona d'aturada, els intervals de clic o els parpellejos necessaris per fer clic dret.

3.2.4 Construcció de l'Interfície Gràfica

Tota construcció software necessita una capa visual. Aquesta capa es va procurar que fos alhora el més unificada possible però també el més oberta possible. Per aconseguir aquest objectiu, es va crear una classe amb mètodes d'interfície a l'estil ModularDispatcher. Aquesta classe es diu: ModularGUI.

La classe proporciona un sistema bàsic per a poder configurar i executar mòduls compatibles amb la Interfície Modular Muse. Proporciona una base visual en la qual assignar diferents Widgets que permetin interactuar amb valors interns per a configurar-los en temps de preexecució. Així mateix, assigna el mòdul configurat a un ModularDispatcher i agrega aquest últim a un servidor osc. També permet assignar controls que modifiquin valors configurables del mòdul en temps d'execució.

ModularGUI necessita rebre un objecte que tingui com a mètodes d'instància: attachConfigurationWidgets, confirm i setRunTimeModuleTools.

- attachConfigurationWidgets permet agregar a una pantalla de configuració, tots aquells inputs o controls necessaris per a configurar el modul desitjat.
- confirm indica que el modul està ja configurat i agrega primerament el mòdul a un ModularDispatcher i aquest a un Servidor. Finalment obra el menú d'execució.
- setRunTimeModuleTools agrega controls al menú d'execució, tant de configuració del modul com de control del servidor osc.

Organitzant les classes de l'interfície d'aquesta forma, s'aspira a estandaritzar el format de menus del sistema pero alhora deixar espai per a les particularitats de cada mòdul.

La principal problemàtica que es va esdevenir dins d'aquest apartat va ser la de controlar i aprendre la llibreria utilitzada Tkinter. Es va triar, ja que és una llibreria standard, però no es tenia massa experiència utilitzant-la.

3.3 Organització i Disseny

Aquest apartat servirà per explicar el projecte a escala estructural i de disseny. Es començarà explicant el sistema de fitxers i la ubicació de les diferents classes dins de cada script, es presentaran uns diagrames de disseny on s'expliqui la interacció dels diferents components entre si i s'acabarà enumerant les diferents llibreries utilitzades i les seves dependències.

3.3.1 Sistema de fitxers

L'estructura de fitxers del projecte es pot veure de manera gràfica en el següent diagrama:

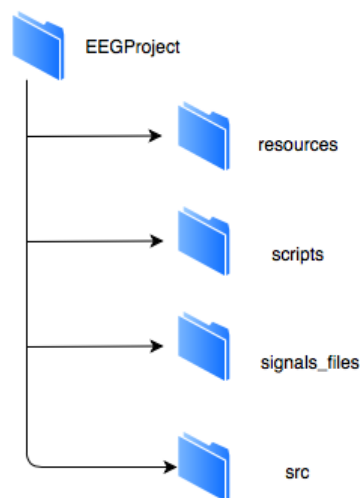


Figura 8. Organització del sistema de fitxers. Font: Elaboració propia

El sistema de fitxers del projecte és simple. La carpeta resources està pensada per contenir tots aquells fitxers o dades extra necessàries per al projecte. La carpeta scripts, està orientada a contenir processos batch necessaris per al bon funcionament del sistema. Singals_files és una carpeta específica del mòdul experimental, que inclou els fitxers de les sessions. El fitxer src inclou tot el codi python del projecte.

Actualment a la carpeta resources hi ha el següent contingut: les llibreries necessaries per a executar el driver muse-io.

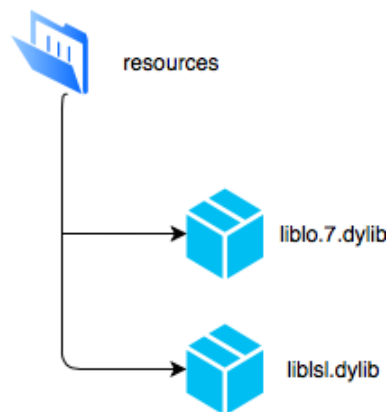


Figura 9. Organització de la carpeta resources. Font: Elaboració propia

Dins la carpeta scripts hi ha el següent contingut: muse-io. El driver necessari per agregar el dispositiu muse a l'ordinador.

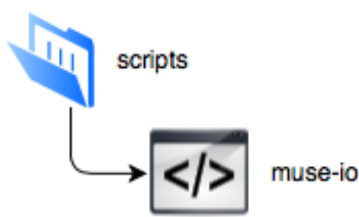


Figura 10. Contingut de la carpeta scripts. Font: Elaboració propia

Finalment, dins la carpeta src hi ha els fixers python que formen el gruix del projecte.

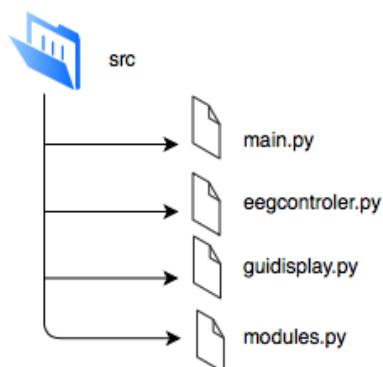


Figura 11. Contingut de la carpeta src. Font: Elaboració propia

El contingut el main.py és infim, simplement inicia el menú d'elecció de modul.

Dins de `eegcontroler.py` hi podem trobar les classes `OSCcontroler` i `ModularDispatcher`. Aquelles necessàries per a iniciar un servidor osc i assignar una interpretació a les senyals.

Per altra banda `guidisplay.py` inclou les classes `ModularGui`, `ExperimentalGui` i `MouseGui`, la classe gràfica base del projecte i les específiques del mòdul que compleixen amb els mètodes d'interfície. També inclou funcions i mètodes complementaris d'ajuda gràfica, així com la funció que inicia el menú d'elecció de mòdul.

Finalment `modules.py` conté els mòduls construïts per al projecte: `MouseModule`, `ExperimentalModule` i `PrintModule`.

3.3.2. Diagrames de disseny

La relació entre els diferents components, drivers i el dispositiu es pot veure reflectida en el següent diagrama:

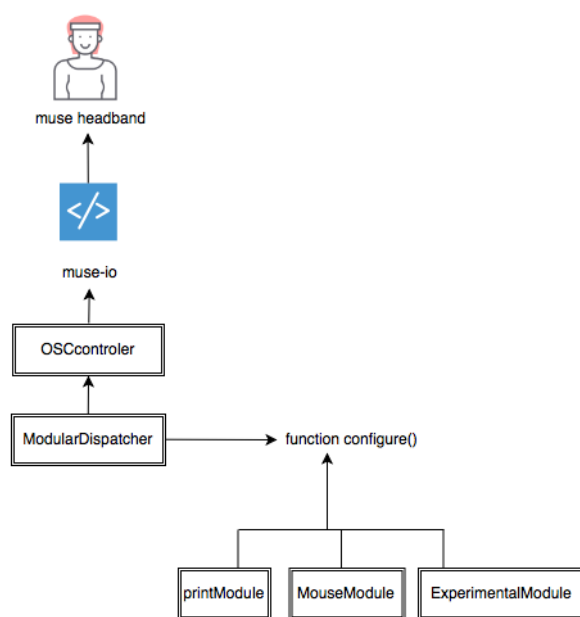


Figura 12. Diagrama general del projecte. Font: Elaboració propia

El dispositiu es comunica amb l'ordinador a través del script `muse-io`. `OSCcontroler` escolta el port on està agregat el dispositiu i captura els senyals. `ModularDispatcher` dona una interpretació als senyals mitjançant la funció `configure` del mòdul que se li ha agregat.

La relació de classes dins entre els components gràfics es pot veure en aquest altre diagrama.

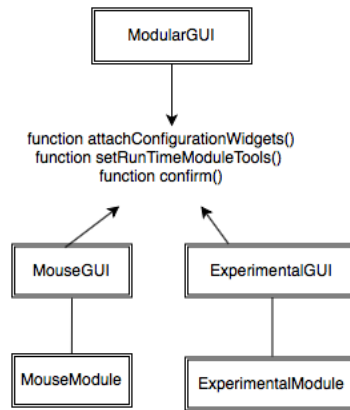


Figura 13. Diagrama de l'interacció entre classes GUI. Font: Elaboració propia

ModularGui és capaç d'interactuar amb tota aquella classe que contingui les funcions d'interfície indicades. Dins de cada classe es pot afegir tants widgets com sigui necessari per al bon funcionament del mòdul amb el qual interactua.

3.3.3 Versions i dependències

El projecte ha estat desenvolupat per a Python 3.6. S'ha procurat que tingués el menor nombre de dependències possible envers una instal·lació més senzilla. La major part de dependències es concentra en els diferents mètodes per interactuar amb elements externs al projecte python. És a dir, amb elements del sistema.

Les llibreries externes a python utilitzades són les següents: pythonosc i pyautogui.

Pyautogui és la única llibreria que té dependències. Per tal de fer-la funcionar cal instal·lar els mòduls pyobjc-core i pyobjc module (en aquest ordre) i el mòdul Pillow.

Pythonosc no té cap dependència, però cal instal·lar-la per a poder fer un ús correcte de tota l'àrea de captura osc.

Bloc 4: Tests amb usuaris

En aquesta quarta secció s'explicarà la sessió d'experiència d'usuari realitzada amb la Interfície Modular Muse agregada amb el mòdul d'accessibilitat. L'objectiu d'aquesta sessió va ser comprovar l'eficàcia en l'ús i l'aprenentatge en usuaris sense cap coneixement previ amb el dispositiu i el sistema d'accessibilitat.

L'experiència va consistir en la realització de tasques complexes però quotidianes, que podria realitzar un usuari típic, amb un ratolí, en un dia qualsevol. Les tasques van estar orientades a simular interaccions dins de l'àmbit de l'oci. Es va voler orientar la sessió dins d'aquest àmbit, ja que normalment no se sol fer en aquest tipus d'experiments.

Es va preparar dues tasques, que tots els participants van realitzar de la mateixa forma. Per tal de medir i comparar l'èxit general, es va cronometrar els temps de cada participant.

A més a més, per tal de servir com a contrast, es van realitzar i cronometrar els temps de les mateixes tasques utilitzant un altre sistema d'accessibilitat. Aquest s'anomena eViacam i està basat en l'ús de la webcam i el reconeixement facial per a moure el cursor.



Figura 14.Usuari utilitzant eViacam

Font: Material audiovisual de la sessió de testeig

4.1 Tasques a realitzar

Com s'ha dit, el tipus de tasques realitzades es podrien catalogar dins de l'àmbit de l'oci. Concretament, la primera tasca a realitzar simulava una interacció típica amb la pàgina de reproducció de vídeos Youtube. La situació inicial amb la qual es trobava l'usuari consistia en un navegador minimitzat, amb una pestanya oberta amb Youtube i un vídeo de música aturat.

La tasca a realitzar contenia els següents passos:

- 1 – Maximitzar el navegador.
- 2 – Engegar el reproductor clicant a la finestra del video.
- 3 – Pausar el video clicant al botó de pausa.
- 4 – Accedir a la pagina principal de Youtube mitjançant l'icona de HOME.
- 5 – Minimitzar el navegador clicant el botó de minimitzar.



Figura 15. Usuari utilitzant l'Interficie Modular.

Font: Material audiovisual de la sessió de testeig

Per altra banda, la segona tasca a realitzar consistia en la interacció amb un joc d'estratègia online per torns anomenat Panzer Marshall. La situació inicial presentada a l'usuari consistia en un navegador minimitzat amb una pestanya oberta dins de la pàgina del joc. El joc estava situat al menú principal sense cap partida iniciada.

Aquesta tasca era més llarga i complexa i contenia els següents passos:

- 1 – Maximitzar el navegador.
- 2 – Clicar sobre el botó *New Game*.
- 3 – Clicar sobre el botó *Campaign*.
- 4 – Clicar sobre el botó *check* dels següents diàlegs.
- 5 – Clicar sobre una unitat d'infanteria per seleccionar-la.
- 6 – Després seleccionar un hexagon proper a un búnker per moure la unitat.
- 7 – Clicar sobre el búnker per atacar-lo.
- 8 – Clicar sobre un avió per seleccionar la unitat.
- 9 – Clicar sobre un enemic per atacar-lo amb l'avió.
- 10 – Minimitzar el navegador.

Així doncs com es pot veure, les tasques eren quotidianes i simulaven una típica tarda avorrida que qualsevol usuari ha experimentat en alguna ocasió. Per veure la documentació original que descriu l'experiència programada es pot consultar l'Annex.

4.2 Metodologia

L'experiència va estar pautaada en diferents parts. Els usuaris que van formar part de l'esdeveniment romanien junts dins d'una sala a part mentre acudien d'un en un a aquella en què vam realitzar l'experiment.

Quan un usuari acudia a la nova sala, se li explicava a grans trets en què consistiria l'experiment i se l'informava que era necessari que firmés un full de conformitat per utilitzar material audiovisual i mètriques extretes de la seva experiència. Després, un cop firmava se li feia omplir un formulari de dades demogràfiques per tal de fer una correlació durant els resultats finals.

Ja satisfetes les formalitats, se li explicava llavors amb detall les tasques i la naturalesa de la sessió. Se li explicava també que realitzaríem primer les tasques amb el software eViacam, després amb el mòdul d'accessibilitat de l'Interfície Modular i finalment, altre cop amb el mòdul d'accessibilitat; però aquest cop s'hauria ensenyat amb més profunditat l'ús del casc als usuaris.



Figura 16. Explicació previa a un usuari abans de la sessió.

Font: Material audiovisual de la sessió de testeig

Es va decidir optar per aquest enfocament per comprovar que l'ús del mòdul d'accessibilitat millorava si s'ensenyava les mecàniques bàsiques. Comprovar aquest fet va sorgir

d'experiències prèvies informals amb diversos usuaris que no acabaven d'entendre o dominar l'ús del casc. Els resultats van ser especialment interessants resolent aquesta qüestió.

Finalment després de l'experiència es va proporcionar un formulari per valorar l'activitat utilitzant els dos sistemes i per deixar comentaris si així es creia oportú.

4.3 Resultats. Anàlisi Quantitatiu

En aquesta secció es tractarà amb el Feedback proporcionat pel grup de voluntaris. El grup d'usuaris constava de 5 persones dels dos gèneres, d'estudis i edat semblants.

En la taula següent es poden veure els seus resultats temporals en minuts, realitzant les tasques programades; tant amb el software eViacam (eV), com la interfície modular Muse (IM), el primer (IM1) i el segon cop (IM2).

	Tasca1(eV)	Tasca2(eV)	Tasca1(IM1)	Tasca2(IM1)	Tasca1(IM2)	Tasca2(IM2)
Persona1	0.59	2.02	2.20	4.37	0.56	1.13
Persona2	1.10	2.36	3.95	6.23	1.05	2.12
Persona3	0.46	2.06	2.57	3.30	1.40	1.58
Persona4	0.48	2.23	3.24	4.56	3.13	3.57
Persona5	0.34	1.56	1.20	3.10	1.24	2.35

Taula 1. Temps d'execució de les tasques segons participant. Font: Elaboració propia (2019)

Les mitjanes doncs es poden veure en la següent taula:

	Mitjana
Tasca1(eV)	0.59
Tasca2(eV)	2.04
Tasca1(IM1)	3,03
Tasca2(IM1)	4,30
Tasca1(IM2)	1,40
Tasca2(IM2)	2,15

Taula 2. Mitjana d'execució de les tasques. Font: Elaboració propia (2019)

Els resultats posen de manifest diverses qüestions. Per començar és evident que la primera tasca resulta més fàcil i es completa en menys temps en tots els casos. La segona tasca és més complexa i tant en un sistema com en l'altre porta més estona de completar.

Comparant eViacam i la Interfície Modular Muse es veu en els resultats que utilitzant el primer software les tasques són més ràpides de realitzar. Això es pot esdevenir per diversos motius.

El principal segurament és a causa de la velocitat en què es mou el cursor utilitzant la Interfície Modular Muse. En aquest sistema la velocitat és bastant més lenta que en eViacam. El segon és perquè el sistema de moviment del mòdul no és del tot intuïtiu.

A diferència d'altre software de control del ratolí, el mòdul no es basa en la direcció en la qual l'usuari mira per moure el cursor. El mòdul utilitza la inclinació relativa del dispositiu respecte a l'eix gravitatori terrestre (mitjançant l'acceleròmetre integrat), per a realitzar el moviment.

Així doncs, per tal de fer correctament els moviments, l'usuari ha d'inclinar el cap a la direcció on vol que es mogui el cursor. Una forma efectiva d'ensenyar els controls als usuaris va ser fer una comparativa amb el moviment que realitza un gos quan està confós. Una altra va ser dir-los que tractessin d'apuntar amb la barbeta i no amb els ulls per tal d'avançar cap a la direcció desitjada.



Figura 17. Gossos confosos inclinant el cap. Font:

<http://la100.cienradios.com/wp-content/uploads/sites/content/2015/11/916030.jpg>

Finalment, en relació amb els temps dels usuaris, es va demostrar que si s'ensenyava a un usuari a fer servir el casc correctament, el seu resultat temporal en qualsevol de les dues activitats podia reduir-se fins, en alguns casos, a la meitat.

4.4 Resultats. Anàlisi Qualitatiu

En aquest apartat es comentaran les diferents valoracions que van puntuar els usuaris en el formulari final.

En la següent taula es pot veure les puntuacions que van assignar els participants als diferents aspectes a valorar dels sistemes dins del formulari. L'esmentat formulari constava de sis preguntes; les tres primeres sobre el software eViacam i les tres últimes sobre el software de Muse.

	Pregunta 1	Pregunta 2	Pregunta 3	Pregunta 4	Pregunta 5	Pregunta 6
Persona1	7	9	5	8	8	10
Persona2	8	8	6	8	6	8
Persona3	8	9	10	4	4	10
Persona4	10	10	10	10	10	10
Persona5	9	10	10	8	6	10

Taula 3. Puntuacions del cuestionari del producte. Font: Elaboració propia (2019)

La primera i la tercera demanaven una valoració general dels 2 softwares. La segona i la cinquena preguntava pel grau de comoditat utilitzant els sistemes. La tercera i la sisena si creia que amb entrenament la seva experiència utilitzant el sistema podria millorar. Per veure l'enunciat i la documentació original es pot consultar l'Annex.

Podem veure les mitjanes observant la taula següent.

	Mitjana
Pregunta 1	8,4
Pregunta 2	9,2
Pregunta 3	8,2
Pregunta 4	7,6
Pregunta 5	6,4
Pregunta 6	9,6

Taula 4. Mitjanes del cuestionari del producte. Font: Elaboració propia (2019)

Veiem doncs que en general la puntuació és més favorable cap al software eViacam. En la pregunta de puntuació general i en la de comoditat guanya a la Interfície Modular però aquesta supera

a eViacam en la pregunta de l'entrenament. Aquest fet fa a relluir el fet que si es realitza entrenament, el software de Muse és molt més fàcil de fer servir.

4.5 Observacions i dades demogràfiques

En aquesta penúltima secció s'exposarà les diferents observacions juntament amb les dades demogràfiques dels participants. Es pot consultar els formularis demogràfics originals en l'Annex. Al ser el primer test d'una versió Alpha, es va escollir usuaris més "propers" al entorn del desenvolupador; per tant, no representen una gran variabilitat demogràfica en relació a la seva edat.

– El **participant 1** és un home d'entre 25 i 45 anys amb un grau d'estudis de Master i exercent dins de l'àrea de treball de la informàtica. No havia utilitzat mai un programa d'accessibilitat.

En el cuestionari sobre els productes comenta:

"Con una mejor explicación de como se usa (ladear la cabeza) mejora muchísimo. Eviacam requiere estar demasiado cerca del monitor o webcam"

– El **participant 2** és una dona d'entre 25 i 45 anys amb un nivell d'estudis d'Enginyeria Informàtica de Sistemes. La seva àrea de treball actual és la consultoria. No havia utilitzat mai cap programa d'accessibilitat.

En el cuestionari sobre els productes comenta:

"Con eViacam hay que tener cuidado con la posición al estar sentado."

– El **participant 3** és un home d'entre 25 i 45 anys amb un nivell d'estudis de tècnic superior. La seva àrea de treball és la informàtica. No havia utilitzat cap sistema d'accessibilitat amb anterioritat.

En el cuestionari sobre els productes comenta:

“eViacam funciona muy fluido y me parece más cómodo, sólo cambiaría el método de clic. Muse es un poco costoso de utilizar, se mueve lento y en general me costaba moverme. En resumen: una fusión de los dos sistemas me parece lo más óptimo, con la fluidez de eViacam y con el sistema de clic de Muse, ya que es más cómodo.”

– El **participant 4** és una dona d'entre 25 i 45 anys amb un nivell d'estudis de llicenciatura en Psicologia. La seva àrea de treball actual és la de la psicologia clínica. No havia utilitzat un software d'accessibilitat amb anterioritat.

En el cuestionari sobre els productes comenta:

“Me ha gustado la interacción. Desde mi punto de vista como profesional sanitaria considero que es un buen desarrollo para mejorar la comodidad y calidad de vida de la persona con diversidad física.”

– El **participant 5** és un home d'entre 18 i 25 anys amb un grau d'estudis de Doctorat. L'àrea de treball actual és la Informàtica. Havia utilitzat amb anterioritat un sistema d'accessibilitat anomenat SnapDragon.

En el cuestionari sobre els productes comenta:

“Ha hagut dificultats per calibrar el Muse. A eViacam, esperar per a fer click. Progressió a l'hora de moure el ratolí a Muse.”

Així doncs, les valoracions personals coincideixen amb els resultats. La Interfície resulta més lenta que eViacam i amb entrenament el seu ús és més fàcil. A més a més veiem un aspecte positiu de la Interfície respecte a eViacam, el sistema de clics, que en dos casos ha resultat més còmode el de la Interfície.

4.6 Conclusions

Després d'aquesta sessió i d'analitzar totes les dades i feedback rebut podem fer un seguit de conclusions.

Conclusió 1: El mòdul d'accessibilitat Muse té una qualitat semblant a un sistema d'accessibilitat actual a disposició del públic.

Encara que eViacam (8,4) ha tingut major puntuació mitjana que el mòdul d'accessibilitat (7,6), podem concloure que la diferència entre els 2 és menor d'un punt, per tant, són d'una qualitat semblant.

Conclusió 2: El mòdul d'accessibilitat Muse té un sistema de clics adequat i eficient.

No hi ha hagut cap comentari negatiu sobre aquest sistema entre els participants i en dues ocasions s'ha considerat que era millor que el de eViacam.

Conclusió 3: Cal millorar la fluïdesa i la comoditat en l'ús del modul d'accessibilitat

Tal com els temps demostren i els comentaris subjectius dels participants deixen a reluir, caldria implementar un control de Muse més veloç. Si es fes, els temps per a realitzar les activitats i les valoracions subjectives segurament millorarien.

Conclusió 4: Amb entrenament, l'ús del mòdul d'accessibilitat millora ostensiblement.

Els resultats demostren que la velocitat i l'eficàcia en l'ús del sistema milloren quan s'ensenya el seu ús. El seu grau de *discoverability* doncs, no és el més elevat, però això no vol dir que el sistema no sigui usable; sinó que en alguns casos pot originar dificultats si no es coneix mínimament.

Conclusió 5: El mòdul MouseModule de l'Interfície Modular Muse és un bon sistema d'accessibilitat

En les preguntes quatre i cinc referents a la qualitat del sistema, en els dos casos es donen puntuacions mitjanes per sobre del 5, per tant podem concloure que el sistema és usable i suficientment eficient per un públic no expert en aquest tipus de sistemes de control. A més, cal tenir en consideració que l'Interfície Modular és una *alpha* desenvolupada per un sol individu que ha competit contra un producte publicat de fa 2 anys per tot un equip.

Bloc 5: Conclusions Finals

Finalment, ara ja només queda fer una retrospectiva sobre tot el treball realitzat sobre el projecte. El producte ha estat el fruit d'una idea i com a tal va generar unes expectatives que s'ha tractat d'assolir. A més, com tot sistema, té punts forts i punts febles i un cop xoca amb la realitat dels usuaris comuns, aquest fet queda rapidament en evidència.

Analitzarem totes aquestes qüestions en els següents punts i finalment s'acabarà fent tot un seguit de propostes per a millorar el producte en treballs futurs.

5.1 Expectatives vs Realitat

Amb el projecte en l'estat actual, s'ha de dir que la majoria d'expectatives que es van plantejar a l'inici del procés de desenvolupament s'han complert. La idea de fer un sistema base per a la interpretació dels senyals és una realitat, i el sistema per a proporcionar un control alternatiu del Mouse també. A més a més, s'ha pogut construir un mòdul d'experimentació bastant eficient.

Per altra banda, l'estat del control del punter no és el que s'esperava a l'inici del projecte. És cert que és eficaç però el moviment resultat més lent de com s'havia plantejat inicialment. A més a més, el control no és del tot intuïtiu i això ha generat problemes en la seva aplicació en usuaris.

Als tests d'usuaris s'ha demostrat aquest fet, però també que és relativament fàcil salvar el problema. Només cal una mica d'entrenament i pràctica.

En definitiva, la majoria d'expectatives s'han complert; tot i així sempre hi ha espai per a millorar.

5.2 Punts forts i punts de millora

La Interfície Modular Muse és un sistema que s'ha tractat de construir el més obert possible. Aquesta és una de les seves principals fortaleeses, que la manipulació dels senyals emesos pel dispositiu Muse, no està tancada només a una sola interpretació. A més, tal com està construïda la Interfície, és capaç de manipular i capturar no només els senyals d'un dispositiu Muse, sinó de tot aquell dispositiu que emeti per OSC.

Referent al mòdul d'experimentació, la seva principal fortaleesa resideix en la tria dels senyals importants i les marques de temps personalitzables durant la sessió de captura. Aquests dos fets poden estalviar molt de treball a un investigador acotant la informació a analitzar.

Un punt de millora d'aquest mòdul seria la divisió en diversos arxius de les dades de captura. Com s'ha comentat en anteriors punts, la mida dels arxius pot arribar a ser massa gros fins hi tot per a pocs minuts depenent del nombre de senyals escollits.

Per altra banda, la principal fortaleesa del mòdul d'accessibilitat resideix en la precisió amb la qual és capaç d'actuar i el seu sistema de clics basat en moviments musculars. Altres sistemes d'accessibilitat no presenten un grau de precisió tan elevat com el d'aquest mòdul i no atorguen un control tan natural dels dos clics del ratolí.

Un punt de millora, seria la velocitat a la qual es mou el ratolí, que durant les sessions de testeig va demostrar ser una mica massa lenta. A més, caldria incloure una secció de tutorials on s'ensenyés els controls de moviment del punter; va quedar demostrat durant la sessió de testeig que el grau de discoverability no era el més òptim, però un cop s'ensenyava els controls del sistema, els usuaris eren capaços de controlar el punter amb molta més eficàcia.

5.3 Treball futur

Finalment, caldria considerar quines són les àrees en què es podria avançar en la millora del producte. Principalment seria en tres aspectes.

El primer aspecte, seria el sistema de menús. Encara que és usable i eficaç, és molt simple i hi ha bastant espai per a la millora. Per exemple, es podria crear una pantalla general en la qual poder canviar de mòdul sense haver de tancar el programa.

Per altra banda, el mòdul d'experimentació captura els senyals a un fitxer però no dóna feedback visual a l'usuari d'aquestes. Un punt a considerar com a treball futur, seria afegir una pantalla al menú que mostres els senyals que s'estan capturant a temps real en gràfica temporal.

Finalment, el mòdul d'accessibilitat també té espai per a la millora i l'evolució. Per una banda caldria afegir unes instruccions o tutorial que mostres a l'usuari com fer servir el casc. Tal com s'ha demostrat al test d'usuaris, milloraria rapidament l'entesa del sistema.

Per altra banda, caldria afegir una zona de velocitat més enllà de la zona de moviment actual. És a dir, el dispositiu disposaria de tres capes de moviment; una zona morta, on el cursor s'aturaria, una zona de moviment lent per a realitzar tasques de precisió i una zona de moviment ràpid més enllà per a desplaçar el cursor amb més velocitat.

Bloc 6: Bibliografia

1. Brain2Speech. (Samin100)

<https://github.com/Samin100/Brain2Speech>

2. Brain Vision Recorder. (Bionic Ibérica)

<https://www.bionic.es/marca/brainvision-recorder/>

3. Camera Mouse. (Boston College)

<http://www.cameramouse.org/index.html>

4. Headmouse. (Universitat de Lleida)

<http://robotica.udl.cat/headmouse.htm>

5. Muse HeadSet. (InteraXon)

<https://choosemuse.com>

6. Muse HeadSet, Documentació per a desenvolupadors. (InteraXon)

<http://developer.choosemuse.com>

7. ProtocolBuffers. (Microsoft)

<https://github.com/protocolbuffers/protobuf/tree/master/python>

8. PyAutoGUI. (Python Software Foundation)

<https://pypi.org/project/PyAutoGUI/>

9. Python-osc. (Python Software Foundation)

<https://pypi.org/project/python-osc/>

10. Spatiotemporal Analysis of Multichannel EEG: CARTOOL. (Denis Brunet, Micah M. Murray, Christoph M. Michel)

<https://www.hindawi.com/journals/cin/2011/813870/>

11. Tendencias del Mercado Laboral. (Michael Page Consultors)

https://www.michaelpage.es/sites/michaelpage.es/files/PG_ER_IT.pdf

ANNEX

Experiencia de Usuario

Durante la siguiente sesión, se propondrán dos casos de uso por los cuales podría necesitarse de un sistema de accesibilidad que proporcionase un control alternativo al del clásico dispositivo externo del ratón, con la misma finalidad y capacidades. El sistema de accesibilidad estaría basado en el mapeo de los movimientos de la cabeza y de diferentes expresiones faciales.

Dichos casos de uso serán ejecutados por el usuario mediante dos sistemas de accesibilidad diferentes, que reúnen los requisitos anteriormente mencionados. Los sistemas de que dispondrá el usuario serán: eViacam y el módulo de accesibilidad de la Modular Muse Interface (MMI).

El usuario deberá tratar de llevar a cabo las tareas propuestas siguiendo el orden establecido en el menor tiempo y con la mayor eficacia posible. Las tareas se realizarán tres veces: una con eViacam, otra con el sistema MMI y otra con el sistema MMI después de entrenar al participante en su uso.

A continuación se describirán los casos de uso propuestos, así como el orden de acciones establecido para llevarlos a cabo.

Caso de uso 1: Youtube

Descripción: El usuario abrirá un video de Youtube mediante un enlace en el navegador. Pausará el video, lo reanudará y luego cerrará el navegador.

Pasos a seguir:

- 1 - Maximizar el navegador.
- 2 - Encender el reproductor clicando en la ventana del video.
- 3 - Pausar el video clicando en el botón de pausa.
- 4 - Acceder a la página principal de Youtube a través del icono de HOME.
- 5 - Minimizar el navegador.

Caso de uso 2: Panzer Marshall

Descripción: El usuario accederá a un juego en línea de estrategia por turnos, moverá unidades y realizará ataques sobre el enemigo, luego cerrará el navegador.

Pasos a seguir:

- 1 – Maximizar el navegador.
- 2 – Pulsar sobre el botón New Game.
- 3 – Pulsar sobre el botón Campaign.
- 4 – Pulsar sobre el botón check de los siguientes dialogos.
- 5 – Pulsar sobre una unidad de infantería para seleccionarla.
- 6 – Después seleccionar un hexágono cercano a un búnker para mover la unidad.
- 7 – Pulsar sobre el búnker para atacarlo.
- 8 – Pulsar sobre un avión para seleccionar la unidad.
- 9 – Pulsar sobre un enemigo para atacarlo con el avión.
- 10 – Minimizar el navegador.

Cuestionario sobre los productos

¿Qué le ha parecido el programa eViacam? Responda del 1 al 10, cuando 1 es muy insatisfactorio y 10 es muy satisfactorio:

¿Se ha sentido cómodo utilizando la aplicación eViacam? Responda del 1 al 10, cuando 1 es muy insatisfactorio y 10 es muy satisfactorio:

¿Cree que con entrenamiento en la aplicación eViacam mejoraría su experiencia? Responda del 1 al 10, cuando 1 es muy en desacuerdo y 10 es muy de acuerdo:

¿Qué le ha parecido la nueva aplicación de accesibilidad para MUSE? Responda del 1 al 10, cuando 1 es muy insatisfactorio y 10 es muy satisfactorio:

¿Se ha sentido cómodo utilizando la nueva aplicación de accesibilidad para MUSE? Responda del 1 al 10, cuando 1 es muy insatisfactorio y 10 es muy satisfactorio:

¿Cree que con entrenamiento en la nueva aplicación de accesibilidad para MUSE mejoraría su experiencia? Responda del 1 al 10, cuando 1 es muy en desacuerdo y 10 es muy de acuerdo:

Observaciones: _____

Cuestionario demográfico

Edad:

- ☐ Menos de 15 años
- ☐ Entre 15 y 18 años
- ☐ Entre 18 y 25 años
- ☐ Entre 25 y 45 años
- ☐ Más de 45 años

Sexo:

- ☐ Mujer
- ☐ Hombre

Nivel de estudios (Completados):

Área de trabajo (Ej. Salud, Servicios, Ingeniería, etc.):

¿Ha utilizado con anterioridad algún programa de accesibilidad? ¿En caso afirmativo, cuál?: _____

Observaciones: _____

CONSENTIMIENTO PARA LA GRABACIÓN DE FOTOGRAFÍAS, VIDEOS y MÉTRICAS
Como participante en este análisis de la experiencia de usuario, entiendo que el uso de la imagen o de la voz del participante, será principalmente para fines de enseñanza, concretamente para el TFG del interesado. Se me informará acerca del uso de la grabación en vídeo o fotografías para cualquier otro fin, diferente a los anteriormente citados.

Aunque para la experiencia se use un Neuroheadset, no se grabará ni usará ningún dato del electroencefalograma, solamente acelerómetros y electromiogramas para detectar las expresiones faciales.

No existe ningún límite de tiempo en cuanto a la vigencia de esta autorización; ni tampoco existe ninguna especificación geográfica en cuanto a dónde se puede distribuir este material. Esta autorización se aplica a las secuencias filmadas en vídeo o fotografías que se puedan recopilar como parte del desarrollo del programa y para los fines que se indican en este documento.

Autorizo a Ferran Martínez Pedreño (Estudiante de la UdL) y a sus tutores a:

☐ Tomar fotografías durante la sesión experimental.

☐ Grabar vídeo durante la sesión experimental.

☐ Una vez anonimizados los datos, usar la información de mi sesión experimental para realizar un estudio estadístico.

*Nombre: _____ *Pasaporte o

DNI: _____

* Edad: _____ E-mail: _____

En Lleida a _____ de _____ de 2019.

Firmado: